

Aplikasi Analisis Biomekanik Untuk Mengembangkan Kemampuan Belari Atlet Lari

Aria Perdana

Jurusan Teknik Informatika, Fakultas Teknologi Industri
Universitas Gunadarma
Jl. Margonda Raya No. 100, Pondok Cina, Depok
Email: perdana.aria@gmail.com

ABSTRAK

Olahraga lari adalah olahraga yang mengandalkan kecepatan. Salah satu faktor untuk mendapatkan kecepatan yang baik adalah teknik berlari yang benar. Untuk mendapatkan teknik berlari yang benar tidaklah mudah. Salah satu cara untuk membantu atlet mendapatkan teknik berlari yang benar adalah dengan melakukan analisis biomekanik terhadap performa atlet. Analisis biomekanik yang paling efektif adalah dengan melakukan analisis terhadap video performa atlet. Sebuah aplikasi dibutuhkan untuk mempermudah analisis tersebut. Aplikasi untuk melakukan analisis ini harus dapat menampilkan suatu hasil pengukuran yang dapat digunakan untuk melakukan analisis dan membantu pengembangan kemampuan berlari atlet. Hasil pengukuran yang didapatkan berupa besar ukuran jarak dan sudut.

Kata Kunci : Grafik, Vektor, Citra, Video, Biomekanik, C#.

PENDAHULUAN

Olahraga adalah sebuah kegiatan yang mengutamakan pengolahan fisik. Olahraga juga merupakan salah satu media untuk membuat kondisi kesehatan manusia menjadi lebih baik dan terjaga.

Dalam olahraga terdapat beberapa cabang, seperti atletik, sepak bola, basket, renang, dan lain-lain. Cabang atletik adalah cabang yang menjadi dasar dari semua cabang olahraga lainnya. Karena bagian-bagian dari atletik dipergunakan dalam cabang olahraga lain, seperti lari, lompatan, dan lemparan.

Dalam cabang olahraga lari terdapat beberapa cabang lagi, seperti lari dengan jarak 100 meter, 200 meter, dan 400 meter. Cabang-cabang olahraga ini adalah cabang olahraga yang sangat populer dan selalu diperlombakan di ajang-ajang lomba internasional. Dalam cabang olahraga ini juga sangat mengutamakan kecepatan, karena kemenangan dalam cabang-cabang ini adalah jika pelari berhasil mencetak waktu yang lebih kecil dari lawan-lawannya, dengan kata lain seorang pelari harus terlebih dahulu sampai ke garis *finish* daripada lawan-lawannya. Untuk mendapatkan kecepatan tidaklah mudah, karena banyak faktor-faktor pendukung untuk mendapatkan kecepatan maksimum dari lari seorang atlet.

Salah satu faktor pendukung atlet untuk mendapatkan kecepatan maksimum adalah seorang atlet harus memiliki teknik berlari yang baik. Pembentukan teknik berlari

yang baik, bisa didapat dengan menjalani latihan secara kontinu, pengawasan dari staff ahli, serta dengan bantuan analisis biomekanik.

Dengan kemajuan teknologi analisis biomekanik bisa dilakukan dengan bantuan komputer. Analisis biomekanik yang dilakukan menggunakan komputer biasanya dilakukan dengan meneliti hasil rekaman sesi latihan atau pertandingan yang dilakukan oleh atlet. Dari hasil analisis biomekanik, data yang didapat berupa kecepatan atlet, sudut-sudut sendi atlet pada saat berlari. Hasil analisis tersebut yang menentukan apakah atlet sudah memiliki teknik berlari yang baik atau tidak.

TINJAUAN PUSTAKA

Biomekanik adalah studi tentang fungsi dan struktural sistem biologi dengan menggunakan metode mekanik. [3]

Pendekatan biomekanik terhadap olahraga lebih difokuskan kepada pelaku olahraga (atlet), tetapi pendekatan biomekanik terhadap olahraga juga bisa merambah kepada perilaku objek yang tidak bergerak seperti alas kaki, permukaan (*field*), dan perlengkapan olahraga yang dapat mempengaruhi performa atlet. [3]

Dasar-dasar dari biomekanik olahraga adalah ilmu mekanika dan sistem anatomi tubuh manusia. Ilmu mekanika sendiri berkaitan erat dengan ilmu hitung, yaitu matematika. Oleh karena itu, pemahaman tentang konsep matematika yang berkaitan dengan ilmu mekanika dan biomekanik sangatlah penting. Konsep matematika tersebut antara lain: persamaan matematika sederhana, kalkulus, serta aljabar vektor.

Analisis biomekanik merupakan perpaduan dari penelitian dan pendekatan teori untuk menjawab pertanyaan-pertanyaan sebagai berikut: Seperti apa teknik berlari yang benar? Pergerakan seperti apa yang baik untuk mendapatkan kecepatan maksimum saat berlari? Tujuan utama analisis biomekanik pada cabang olahraga lari adalah untuk mengurangi cedera dan meningkatkan kemampuan atlet. [4]

Teknik-teknik dalam pengumpulan data dan pemrosesan data sangatlah penting dalam analisis biomekanik. Salah satu teknik untuk keperluan tersebut adalah menggunakan video dan cinematografi dua atau tiga dimensi.

Analisis biomekanik dikategorikan menjadi dua jenis, yaitu: analisis kualitatif dan kuantitatif. Analisis kualitatif berkaitan dengan deskripsi perilaku olahraga yang subjektif dan teliti; sedangkan analisis kuantitatif membutuhkan pengukuran dan hasil data yang detil [3]. Analisis kualitatif harus dilakukan secara lebih dahulu sebelum analisis kuantitatif untuk mendapatkan pengertian yang komprehensif pada variabel yang akan diukur.

Teknik paling utama dalam merekam dan mempelajari biomekanik adalah video dan cinematografi. Keunggulan dari dua teknik ini adalah bisa merekam performa atlet di mana saja, bukan hanya di laboratorium penelitian. Kedua teknik ini juga dapat meminimalisasi gangguan yang mungkin terjadi pada atlet ketika berlari.

Citra

Data atau informasi tidak hanya dapat disajikan dalam bentuk teks, tetapi juga dapat berupa gambar. Citra atau gambar memegang peranan penting sebagai bentuk informasi visual. Citra memiliki karakteristik yang tidak dimiliki oleh data teks, yaitu citra kaya akan informasi.

Secara harfiah, citra adalah gambar pada bidang dua dimensi. Ditinjau dari sudut pandang matematis, citra merupakan fungsi *continue* dari intensitas cahaya pada bidang dua dimensi [9]. Sumber cahaya menerangi objek, kemudian objek memantulkan kembali sebagian dari berkas cahaya tersebut ke alat-alat optik, seperti mata manusia, kamera, dan lain-lain, sehingga terbentuklah bayangan objek pada alat-alat optik tersebut, yang kemudian bayangan itu disebut citra terekam.

Citra dikategorikan menjadi dua jenis, yaitu citra diam dan citra bergerak. Citra diam adalah citra tunggal yang tidak bergerak. Citra bergerak adalah rangkaian citra diam yang ditampilkan secara sekuensial, sehingga memberi kesan sebagai gambar yang bergerak. Setiap citra di dalam rangkaian itu disebut *frame*.

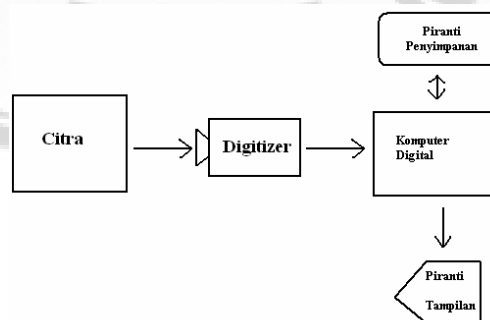
Citra harus direpresentasikan secara numerik dengan nilai-nilai diskrit agar dapat diolah dengan komputer digital. Representasi citra dari fungsi kontinu, menjadi nilai-nilai diskrit disebut digitalisasi. Citra yang dihasilkan melalui digitalisasi ini disebut citra digital [9]. Pada umumnya citra digital berbentuk empat persegi panjang, dan dimensi ukurannya dinyatakan sebagai tinggi X lebar atau lebar X panjang.

Proses digitalisasi citra ada dua macam:

1. Digitalisasi spasial (x,y), yang biasa disebut sebagai sampling.
2. Digitalisasi intensitas $f(x,y)$, yang biasa disebut sebagai kuantisasi.

Secara umum, elemen yang terlibat dalam digitalisasi citra dapat dibagi menjadi empat komponen, yaitu:

- a. digitizer
- b. komputer digital
- c. piranti tampilan
- d. piranti penyimpanan



Gambar 1: Elemen Digitalisasi Citra

Digitizer merupakan sistem penangkap citra digital yang melakukan penjelajahan citra dan mengkonversinya menjadi representasi numerik sebagai masukan untuk komputer digital. Hasil dari *digitizer* adalah matriks yang elemen-elemennya menyatakan nilai intensitas cahaya pada suatu titik.

Digitizer terdiri dari tiga komponen dasar yaitu: sensor citra yang bekerja sebagai pengukur intensitas cahaya, perangkat penjelajah yang berfungsi merekam hasil pengukuran intensitas pada seluruh citra, dan pengubah analog ke digital yang berfungsi melakukan penerokan dan kuantisasi [9]. Contoh dari *digitizer* adalah kamera digital dan *scanner*.

Bitmap

Bitmap adalah representasi dari citra grafis yang terdiri dari array dua dimensi, di mana setiap elemen array merepresentasikan warna yang disimpan pada lokasi tertentu [8]. Saat ditampilkan pada layar komputer, setiap elemen digambarkan menjadi sebuah pixel tunggal. Jika pixel berada terlalu dekat dengan piranti tampilan, maka akan sulit bagi mata manusia untuk mendeteksi struktur array yang membentuk citra tersebut.

Citra bitmap pada windows biasa disebut dengan sebutan Windows BMP. Windows BMP adalah format citra asli untuk sistem operasi Microsoft Windows [8]. Windows BMP mendukung citra dengan 1, 2, 4, 8, 16, 24, dan 32 bit per pixel. Citra dalam format BMP ada tiga macam: citra biner, citra berwarna, dan citra hitam-putih.

Pada setiap berkas BMP terdiri atas header berkas, header bitmap, informasi palet, dan data bitmap [8].

Header berkas	Header bitmap	Informasi Palet	Data bitmap
14 byte	12 s/d 64 byte	0 s/d 1024 byte	N byte

Gambar 2: Struktur File BMP

Video

Video adalah teknologi yang berfungsi untuk merekam, menangkap, memroses, menyimpan, mentransmisi, dan merekonstruksi kumpulan citra diam secara sekuensial, sehingga dapat merepresentasikan *scene* yang bergerak. [1]

Istilah video biasanya digunakan untuk mendeskripsikan format penyimpanan untuk citra bergerak: format video digital, termasuk DVD, MPEG-4, avi, dan QuickTime; video analog, termasuk VHS dan Betamax. Video dapat direkam dan ditransmisikan melalui berbagai media fisik: di pita magnetik saat direkam melalui kamera video sebagai sinyal elektrik PAL atau NTSC, atau sebagai MPEG-4 atau DV saat direkam menggunakan kamera digital. [1]

Frame rate adalah jumlah citra diam yang muncul per detik saat video dimainkan, sehingga menimbulkan ilusi bahwa subjek di dalam layar tersebut bergerak. [13]

Frame rate biasanya disingkat menjadi fps. Setiap standar sinyal elektrik seperti PAL atau NTSC memiliki frame rate yang berbeda.

Tabel 1: Perbedaan Frame Rate Pada Standar Sinyal Elektrik Video

No	Sinyal Elektrik	Frame Rate
1	NTSC	29,97 fps
2	PAL	25 fps

Video sebenarnya merupakan kumpulan citra diam (frame) yang dimainkan secara sekuensial. [13]

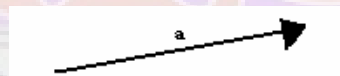
Setiap frame memiliki sebuah angka unik untuk mengenali fram tersebut yang disebut dengan *timecode*. Semua durasi dan lokasi akan terjadinya perubahan pada sebuah frame, akan menunjuk *timecode* menjadi titik acuan. Timecode biasanya dituliskan dalam bentuk:

jam : menit : detik : frame

Pada penulisan bentuk *timecode* di atas, frame bukan digunakan sebagai sebuah unit pengukur, melainkan merujuk pada sebuah citra diam yang menjadi bagian dari video. Bagian frame selalu dimulai dengan angka 00 dan akan bertambah sampai dengan frame rate dari video. Sebagai contoh, pada NTSC frame akan dihitung dari 00 sampai dengan 29.

Vektor

Vektor merupakan suatu ruas garis yang memiliki panjang dan arah [6]. Suatu vektor digambarkan sebagai garis dengan anak panah untuk menunjukan arah [6].



Gambar 3: Contoh Vektor

Vektor berhubungan erat dengan letak sebuah titik pada sebuah bidang. Letak titik pada bidang disajikan di dalam sistem koordinat, yang biasanya berupa sistem koordinat Cartesius, tetapi tidak menutup kemungkinan titik tersebut disajikan pada sistem koordinat lainnya. Titik yang terletak pada sistem koordinat berkorespondensi dengan pasangan bilangan berurut (x , y). Pasangan bilangan berurut ini merupakan koordinat bidang titik, yang dapat digunakan untuk menentukan jarak serta arah vektor terhadap sumbu- x dan sumbu- y .

Setiap vektor memiliki sebuah *magnitude* (setara dengan panjang atau ukuran). Panjang tersebut dapat didapatkan dengan perhitungan sebagai berikut. Jika suatu vektor direpresentasikan oleh n -tupel ($x_1, x_2, x_3, \dots, x_n$), maka panjang vektor a yang merepresentasikan panjang vektor dari ekor sampai ke kepala vektor adalah [6]:

$$|a| = \sqrt{x_1^2 + x_2^2 + x_3^2 + \dots + x_n^2}$$

Panjang vektor bisa digunakan untuk menghitung perkalian skalar. Perkalian skalar atau *dot product* antara vektor a dan vektor b dilambangkan dengan $a \cdot b$ (a dot b) menghasilkan sebuah bialangan real. Perkalian skalar itu didefinisikan sebagai [2]:

$$a \cdot b = |a| |b| \cos \theta \text{ jika } a \neq 0 \text{ dan } b \neq 0$$

dan
 $\mathbf{a} \cdot \mathbf{b} = 0$ jika $\mathbf{a} = \mathbf{0}$ dan $\mathbf{b} = \mathbf{0}$

Dimana $|\mathbf{a}|$ dan $|\mathbf{b}|$ masing-masing menyatakan panjang vektor \mathbf{a} dan \mathbf{b} . θ menyatakan sudut yang dibentuk oleh vektor \mathbf{a} dan \mathbf{b} . Nilai $\mathbf{a} \cdot \mathbf{b}$ juga akan menjadi nol apabila besar θ dengan 90 derajat.

Dot product dari vektor $\mathbf{a} (x_1 \ x_2)$ dengan vektor $\mathbf{b} (y_1 \ y_2)$ juga bisa dihitung dengan persamaan [2]:

$$\mathbf{a} \cdot \mathbf{b} = x_1y_2 + x_2y_1$$

METODE PENELITIAN

Metode penelitian untuk membuat aplikasi analisis biomekanik ini dilakukan dengan cara melakukan analisis terhadap permasalahan inti, dan kemudian dilanjutkan dengan membuat perancangan untuk memecahkan masalah tersebut.

Analisis Permasalahan

Pada program pengembangan kemampuan atlet lari, ada banyak pertanyaan yang harus dicari jawabannya seperti: “Seperti apa teknik berlari yang benar?” atau “Pergerakan seperti apa yang baik untuk mendapatkan kecepatan maksimum saat berlari?”. Pertanyaan-pertanyaan tersebut merupakan permasalahan yang dapat dipecahkan dengan bantuan analisis biomekanik.

Seperti yang sudah disebutkan pada bab sebelumnya bahwa analisis biomekanik memiliki tujuan utama yaitu mengurangi cedera dan meningkatkan kemampuan atlet. Oleh karena itu, analisis biomekanik merupakan elemen penting pada program pengembangan kemampuan atlet.

Teknik terbaik untuk melakukan analisis biomekanik adalah dengan menggunakan video, sehingga dibutuhkan alat bantu untuk melakukan analisis biomekanik. Alat bantu dapat berupa kamera video dan komputer yang memiliki aplikasi untuk melakukan analisis biomekanik. Aplikasi yang dibutuhkan adalah aplikasi yang dapat mengukur sudut-sudut dan jarak yang terdapat pada atlet, ketika atlet sedang berlari. Dengan adanya aplikasi seperti ini, diharapkan dapat membantu program pengembangan kemampuan atlet lari.

Perancangan Aplikasi

Perancangan aplikasi dipecah ke dalam beberapa tahap. Tahap-tahap tersebut adalah sebagai berikut:

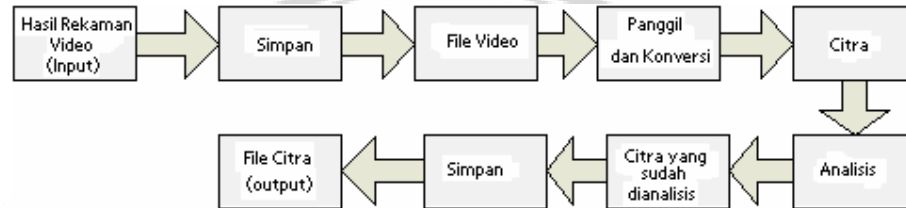
1. Perancangan konsep awal atau gambaran umum.
2. Perancangan fungsi utama aplikasi yang menyangkut konversi video menjadi citra diam dan perancangan proses pengukuran untuk analisis biomekanik.
3. Perancangan system aplikasi.

1. Gambaran umum aplikasi

Aplikasi analisis biomekanik adalah aplikasi yang berfungsi untuk melakukan analisis biomekanik seorang atlet. Analisis dilakukan untuk membantu pengembangan kemampuan seorang atlet. Aplikasi ini dibuat untuk atlet cabang lari atau atletik.

Pada aplikasi ini, analisis biomekanik dilakukan pada sebuah citra diam yang didapat dari video rekaman performa atlet. Tahapan-tahapan yang dilewati untuk melakukan analisis adalah:

1. Perekaman video;
2. Penyimpanan file video ke dalam media penyimpanan;
3. Pemanggilan file video oleh aplikasi dan mengubahnya menjadi citra diam;
4. Analisis menggunakan bantuan garis-garis yang digambarkan di atas citra diam tersebut;
5. Penyimpanan hasil analisis berupa citra ke dalam media penyimpanan.



Gambar 4: Tahapan Proses Aplikasi

Dari tahapan-tahapan yang sudah disebutkan, dapat disimpulkan bahwa inputan terhadap aplikasi ini berupa file video dan output berupa citra. Analisis dilakukan dengan cara menghitung sudut dengan bantuan garis dan vektor. Pada proses pengukuran sudut digunakan aljabar vektor, sedangkan untuk menghitung jarak akan digunakan rumus perbandingan skala.

2. Perancangan konsep konversi video menjadi citra diam

Analisis biomekanik akan dilakukan terhadap sebuah citra yang diam. Tetapi, inputan untuk aplikasi ini berupa sebuah video. Untuk itu harus dilakukan sebuah proses konversi video menjadi citra diam.

Pada dasarnya, video merupakan kumpulan citra diam yang dimainkan secara sekuensial sehingga memperlihatkan bahwa objek di dalam video bergerak. Oleh karena itu citra diam yang ada di dalam video bisa diekstrak untuk dilihat secara satu persatu.

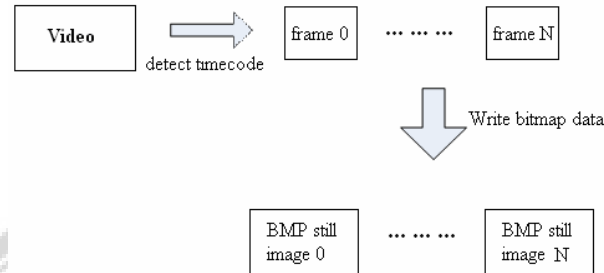
Setiap citra diam atau frame memiliki sebuah *timecode* yang unik. *Timecode* ini digunakan untuk mengenali citra diam yang membentuk sebuah video. Untuk mengonversi sebuah video menjadi citra diam, maka *timecode* harus dideteksi terlebih dahulu. Apabila *timecode* sudah terdeteksi, maka informasi seperti berapa panjang video dan berapa banyak citra yang membentuk video tersebut dapat diketahui.

Apabila informasi tersebut sudah diketahui, maka untuk pemanggilan citra diam akan menjadi lebih mudah. Tetapi, bentuk dari citra diam yang ingin ditampilkan masih belum diketahui. Oleh karena itu, pembentukan format citra diam harus dilakukan.

Format citra diam bitmap merupakan format citra asli yang memiliki kualitas yang bagus. Bitmap juga merupakan format citra yang paling mudah untuk dibaca dan ditulis. Seperti yang sudah dibahas sebelumnya bahwa format citra bitmap memiliki struktur data yang terdiri dari header berkas, header bitmap, informasi palet, dan data bitmap. Untuk membentuk frame menjadi sebuah bitmap, dapat dilakukan dengan

menuliskan data-data frame yang sudah dikenali dan menuliskannya menjadi data bitmap untuk citra yang ditampilkan.

Jika data bitmap sudah ditulis, maka bitmap sudah terbentuk dan citra tersebut dapat ditampilkan.



Gambar 5: Proses Konversi Video

3. Perancangan proses pengukuran untuk analisis biomekanik

Untuk perancangan proses pengukuran dilakukan dalam beberapa tahap. Tahap-tahap yang harus dilalui adalah:

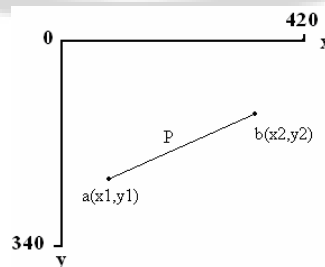
a. Penetapan resolusi citra

Tahap awal proses biomekanik, hal yang pertama harus dilakukan adalah mengubah video menjadi citra beresolusi tetap. Resolusi citra harus tetap untuk memudahkan pengukuran saat melakukan analisis.

Untuk resolusi citra akan ditetapkan sebesar 420 X 340. Besar resolusi ini cukup untuk melakukan analisis karena resolusi citra cukup besar. Jika resolusi terlalu kecil ataupun terlalu besar, akan berdampak kurang bagus terhadap citra.. Misalnya saja, jika resolusi terlalu kecil akan menyebabkan kesulitan dalam melakukan analisis terhadap citra, sedangkan resolusi yang terlalu besar akan menghasilkan citra yang buram atau pecah.

b. Menggambar garis di atas citra hasil konversi

Untuk melakukan analisis biomekanik, dibutuhkan alat bantu untuk melakukan beberapa pengukuran. Dalam pembuatan aplikasi analisis biomekanik ini, faktor analisis yang ingin diukur adalah jarak dan sudut. Untuk melakukan pengukuran tersebut akan dibantu oleh garis-garis yang akan digambar atau ditambahkan di atas citra. Garis-garis tersebut dibuat dengan menghubungkan dua atau lebih titik-titik yang memiliki koordinat tertentu.



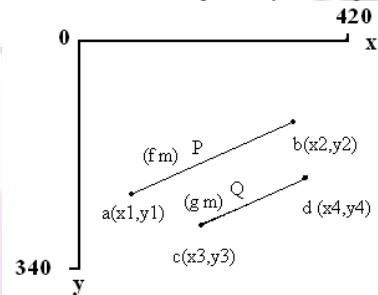
Gambar 6: Garis Dengan Titik Koordinat

c. Mengukur jarak yang terdapat di dalam citra

Untuk mengukur jarak, dibutuhkan garis yang menghubungkan dua buah titik koordinat. Pada gambar 3.4 dapat dilihat terdapat sebuah garis yang menghubungkan titik a dengan titik b. Garis ini memiliki panjang yang dapat diperoleh dengan menggunakan persamaan:

$$P = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

Panjang garis tersebut akan didapat dengan satuan pixel. Dari persamaan di atas, maka garis ab memiliki panjang sebesar P pixel. Panjang garis ini sangat berguna untuk mengukur jarak dengan cara membandingkannya dengan garis yang lain.



Gambar 7: Perbandingan Dua Buah Garis

Pada gambar 7 dapat dilihat adanya dua buah garis. Setiap garis memiliki panjang yang berbeda. Garis ab memiliki panjang sebesar P pixel, sedangkan garis cd memiliki panjang Q pixel. Untuk garis ab yang memiliki panjang sebesar P pixel, garis tersebut juga mewakili besaran panjang dalam satuan meter sebesar f meter, sehingga akan menghasilkan sebuah perbandingan skala:

$$1 : f \text{ (m)} = 1 : P \text{ (pixel)} \dots\dots\dots(1)$$

Jadi setiap satu pixel garis ab akan mewakili P/f meter. Persamaan satu juga dapat diterapkan pada garis cd dan mengukur panjang. Untuk mengukur panjang, salah satu garis dijadikan garis referensi yang nantinya akan mewakili sebuah ukuran yang statis yang terdapat pada citra. Jika garis ab yang menjadi referensi, maka nilai P dan f sudah diketahui. Nilai f akan mewakili ukuran statis yang terdapat pada citra nantinya. Ukuran statis yang diketahui dapat berupa lebar track, panjang track, atau objek lainnya yang memiliki ukuran tetap dengan Menggunakan garis ab sebagai referensi dan menggunakan perbandingan skala, maka untuk mendapatkan nilai g digunakan persamaan sebagai berikut:

$$f : P = g : Q \dots\dots\dots(2)$$

Nilai f, P, dan Q sudah diketahui, hanya nilai g yang tidak diketahui. Maka untuk mencari nilai g persamaan dua diubah menjadi:

$$g = (f * Q) / P \dots\dots\dots(3)$$

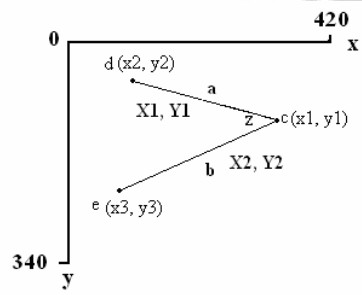
Dari hasil perhitungan menggunakan persamaan tiga, maka besar nilai g akan menjadi sebesar $(f * Q)/P$ meter. Untuk pengukuran jarak hanya dibutuhkan satu referensi, sedangkan garis yang menjadi indikator jarak dapat digambar lebih dari satu.

d. Mengukur sudut terhadap atlet di dalam citra

Berbeda dengan mengukur jarak, untuk mengukur sudut dibutuhkan dua buah garis yang saling terhubung di sebuah titik dan memiliki koordinat sebanyak tiga buah. Untuk mengukur sudut tidak dapat dilakukan dengan perbandingan skala, akan tetapi harus dilakukan dengan menggunakan aljabar vektor. Aljabar vektor yang digunakan untuk mengukur sudut adalah rumus perkalian skalar vektor. Rumus perkalian skalar vektor yang akan digunakan adalah:

$$\mathbf{a} \cdot \mathbf{b} = |\mathbf{a}| |\mathbf{b}| \cos \theta \dots\dots(4)$$

Huruf **a** dan **b** mewakili dua buah vektor yang berbeda, akan tetapi berpangkal pada sebuah titik koordinat yang sama. Sedangkan **|a|** dan **|b|** mewakili panjang vektor **a** dan **b**.



Gambar 8: Dua Buah Vektor Yang Membentuk Sudut

Pada gambar 8 dapat dilihat terdapat dua buah vektor **a** dan **b** yang berpangkal pada titik **c** yang berkoordinat pada titik **x1** dan **y1**. Untuk menggunakan rumus perkalian skalar vektor, nilai **a . b** harus diketahui. Nilai **a . b** bisa di dapatkan dengan menggunakan rumus yang merupakan turunan dari persamaan 4:

$$\mathbf{a} \cdot \mathbf{b} = X1 * X2 + Y1 * Y2 \dots\dots(5)$$

Di mana nilai **X1**, **X2**, **Y1**, dan **Y2** bisa didapat dengan persamaan-persamaan sebagai berikut:

$$X1 = |x2 - x1| \dots\dots(6)$$

$$X2 = |x3 - x1| \dots\dots(7)$$

$$Y1 = |y2 - y1| \dots\dots(8)$$

$$Y2 = |y3 - y1| \dots\dots(9)$$

Apabila nilai **a . b** sudah diketahui, maka besar panjang vektor **a** dan **b** harus diketahui. Untuk besar panjang vektor bisa didapat dengan menggunakan rumus sebagai berikut:

$$|\mathbf{a}| = \sqrt{(X1^2 + Y1^2)} \dots\dots(10)$$

$$|\mathbf{b}| = \sqrt{(X2^2 + Y2^2)} \dots\dots(11)$$

Di mana nilai **X1**, **X2**, **Y1**, dan **Y2** adalah nilai yang didapat dari persamaan 6, 7, 8, dan 9. Apabila semua nilai yang dibutuhkan sudah didapat, maka pengukuran besar sudut yang diwakili oleh notasi **Z** pada gambar bisa dilakukan. Pertama kita gunakan persamaan rumus perkalian skalar vektor yang notasinya disesuaikan dengan notasi yang terdapat pada gambar 8.

$$\mathbf{a} \cdot \mathbf{b} = |\mathbf{a}| * |\mathbf{b}| * \cos Z \dots\dots(12)$$

Karena persamaan ini digunakan untuk mengukur sudut, maka nilai $\cos Z$ adalah nilai yang ingin dicari. Oleh karena itu maka persamaan 12 diubah menjadi:

$$\cos Z = a \cdot b / |a| \cdot |b| \dots\dots\dots(13)$$

Setelah persamaan 13 didapat, maka nilai $\cos Z$ dapat dicari. Berikutnya adalah menurunkan persamaan 13 dengan memasukkan nilai a , b , $|a|$, dan $|b|$ yang sudah dicari menggunakan persamaan 5, 10, dan 11 menjadi:

$$\cos Z = (X1 \cdot X2 + Y1 \cdot Y2) / (\sqrt{X1^2 + Y1^2}) \cdot \sqrt{X2^2 + Y2^2}) \dots\dots\dots(14)$$

Dari persamaan 14, nilai $\cos Z$ akan di dapat sebagai sebuah bilangan real. Besar sudut Z bisa didapat dengan fungsi trigonometri:

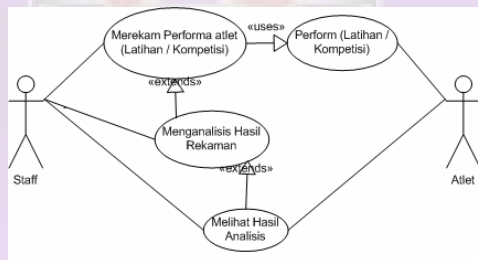
$$\cos^{-1} Z \text{ atau } \text{ArcCos } Z$$

4. Perancangan system aplikasi

Perancangan sistem untuk aplikasi analisis biomekanik, yang bertujuan mengembangkan kemampuan berlari atlet lari ini, menggunakan UML (*Unified Modelling Language*). Diagram-diagram yang digunakan dalam perancangan tersebut adalah *use case diagram*, *class diagram*, dan *activity diagram*.

a. Perancangan Diagram Use Case

Diagram ini digunakan untuk menggambarkan pengguna aplikasi dan perilaku pengguna yang berkaitan dengan sistem yang berpengaruh terhadap pembuatan aplikasi. Diagram *use case* ini menceritakan skenario yang berkaitan dengan penggunaan aplikasi secara global [10]. Pelaku dalam diagram ini disebut aktor, sedangkan perilaku yang dimiliki oleh aktor disebut *use case* [10].

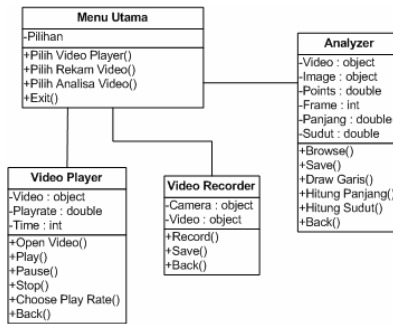


Gambar 9: Diagram Use Case

Diagram *use case* pada gambar 9 menceritakan kegiatan sistem yang berkaitan dengan aplikasi yang akan dibuat. Terdapat dua aktor yaitu Staff dan Atlet. Kedua aktor tersebut memiliki perilaku masing-masing yang saling berhubungan.

b. Perancangan Class Diagram

Class diagram digunakan untuk menggambarkan objek-objek, operasi objek, dan relasi yang menghubungkan objek-objek yang terdapat dalam sistem. Objek di diagram ini didefinisikan sebagai *class*, dan objek tersebut memiliki atribut dan operasi. [10]



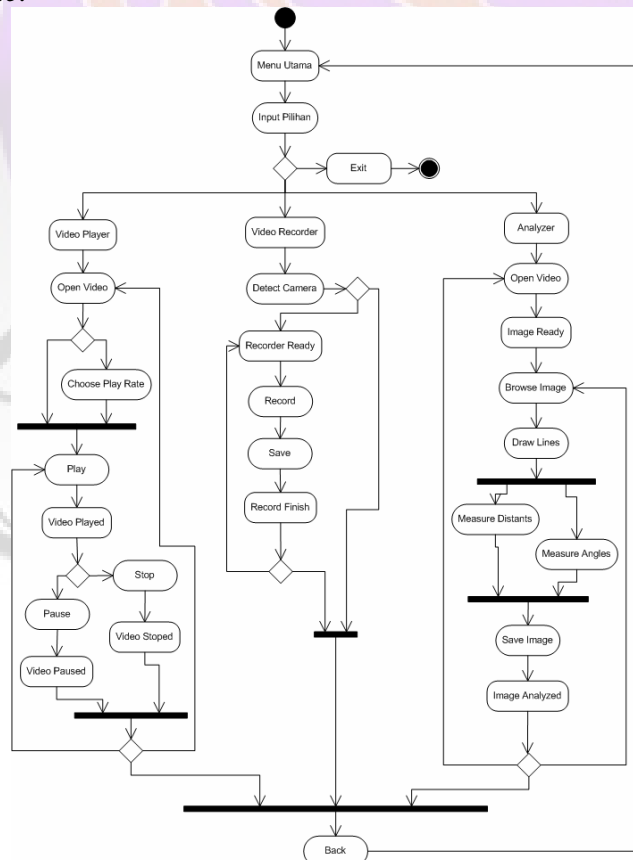
Gambar 10: Diagram Class

Diagram *class* pada gambar 10 menggambarkan objek-objek yang membentuk aplikasi. Kelas-kelas pada diagram ini akan diimplementasikan menjadi *form* pada tahap pembuatan aplikasi. Terdapat empat buah kelas pada diagram ini, yaitu: Menu Utama, Video Player, Video Recorder, dan Analyzer.

c. Perancangan Diagram Activity

Perancangan menggunakan diagram *activity* adalah untuk menggambarkan logika prosedural yang akan dijalankan oleh sistem dan aplikasi. Diagram *activity* memiliki peran yang mirip dengan *flowchart*. [10]

Diagram *activity* digunakan untuk menjelaskan alur aplikasi secara keseluruhan. Diagram ini akan diawali oleh sebuah *initial state* dan akan selalu diakhiri dengan sebuah *final state*.



Gambar 11: Diagram Activity Aplikasi

HASIL DAN PEMBAHASAN

Rancangan dan analisis pada penulisan ini akan diterapkan ke dalam sebuah pembuatan aplikasi. Pembuatan aplikasi akan menggunakan bahasa pemrograman Visual C#. Aplikasi akan dibuat sesuai dengan analisis dan rancangan yang sudah ada. Sesuai dengan hasil perancangan maka form-form yang dihasilkan antara lain:

a. Form Menu Utama

Form menu utama berfungsi untuk menjadi pusat pemilihan form lain yang tersedia di dalam aplikasi dan juga sebagai form utama aplikasi. Karena form menu utama adalah form yang pertama kali muncul saat aplikasi akan dijalankan. Terdapat tiga pilihan form pada aplikasi ini, yaitu Video Player, Video Recorder, dan Analyzer. Form ini dibuat sesederhana mungkin untuk mempermudah pemilihan dan tidak membingungkan.



Gambar 12: Tampilan Form Menu Utama

b. Form Video Player

Form ini berfungsi untuk memainkan file video. Form ini berfungsi untuk menganalisis performa atlet secara langsung tanpa melakukan pengukuran. Analisis yang dilakukan berupa review video rekaman performa atlet, dan video tersebut dapat dimainkan secara gerakan lambat atau *slow motion*.

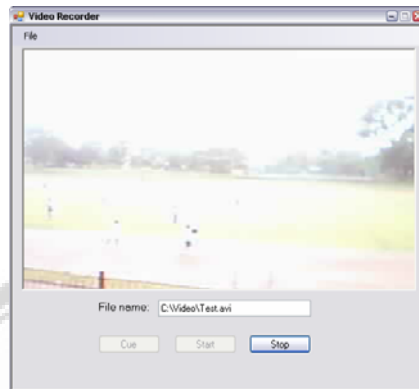


Gambar 13: Tampilan Form Video Player Saat Uji Coba

c. Form Video Recorder

Form ini berfungsi untuk merekam video performa atlet, baik saat atlet latihan atau berlomba. Hasil dari form ini merupakan video dengan extension avi dan didapat sebuah file video bernama test.avi. Pada saat uji coba, video yang didapat dari form ini kurang bagus. Hal ini disebabkan penggunaan kamera yang beresolusi kecil dan

tidak cocok digunakan di luar ruangan. Jadi hasil yang didapat dari form ini sangat bergantung dengan jenis kamera yang digunakan.

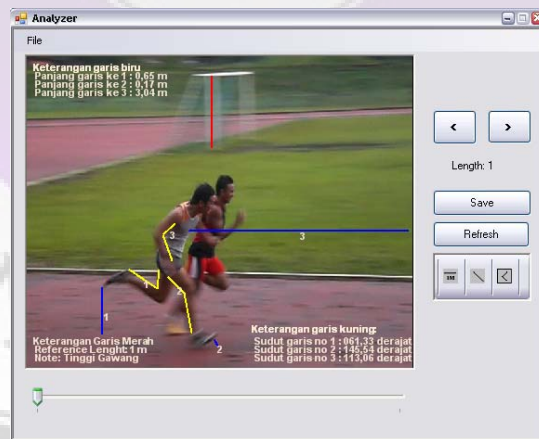


Gambar 14: Tampilan Form Video Recorder Saat uji Coba

d. Form Analyzer

Form analyzer merupakan form untuk melakukan analisis biomekanik. Analisis biomekanik dilakukan dengan cara mengukur sudut dan jarak yang terdapat di dalam citra yang ditampilkan. Form ini merupakan form yang paling kompleks di antara form yang lainnya. Bisa dikatakan, form ini merupakan inti dari aplikasi yang dibuat.

Seperti yang sudah disebutkan sebelumnya, analisis dilakukan terhadap citra diam yang didapat dari sebuah file video. Dalam hal ini file video bisa merupakan hasil rekaman melalui form video recorder atau kamera yang tidak terhubung dengan aplikasi. Video yang dianalisis diambil dari media penyimpanan terhubung pada komputer.



Gambar 15: Tampilan Form Analyzer Saat Uji Coba

Dari form ini pula *output* dari aplikasi didapatkan. *Output* merupakan sebuah citra diam dengan format bitmap. Citra tersebut memiliki resolusi sebesar 420 x 340.



Gambar 16: Output Aplikasi

Dari gambar 16 dapat dilihat beberapa keterangan yang tertulis pada pojok-pojok citra tentang hasil analisis. Garis merah yang digambarkan di atas sebuah objek statis (tiang gawang) yang berukuran tetap dijadikan sebagai referensi. Dalam citra tinggi tiang gawang tersebut dituliskan sepanjang 1 m.

Garis biru yang digambarkan di sekitar atlet yang sedang berlari, merupakan garis yang berfungsi untuk mengukur jarak yang ingin diketahui. Apabila dilihat pada gambar 16 maka jarak-jarak tersebut adalah tinggi angkatan kaki kiri dan kanan, serta jarak dari atlet sampai pinggiran citra yang ada di depan atlet. Setiap garis diberi nomor supaya tidak tertukar dengan keterangan yang tertulis.

Garis kuning merupakan garis untuk mengukur sudut. Untuk pengukuran sudut tidak perlu adanya referensi. Jika diamati pada gambar 16, sudut yang diukur adalah sudut kaki kiri, kaki kanan, dan tangan kanan. Sama seperti grafik yang lain, grafik ini juga memiliki keterangan yang diletakan di pojok citra ini.

Tabel 2: Rangkuman Hasil Analisis

No	Warna Garis	No Garis	Nilai	Keterangan
1	Merah	-----	1 m	Tinggi tiang gawang
2	Biru	1	0,65 m	Tinggi angkatan kaki kiri
3	Biru	2	0,17 m	Tinggi angkatan kaki kanan
4	Biru	3	3,04 m	Jarak dengan sisa citra (<i>track</i> yang terlihat)
5	Kuning	1	61,33 derajat	Sudut kaki kiri
6	Kuning	2	145,54 derajat	Sudut kaki kanan
7	Kuning	3	113,06 derajat	Sudut tangan kanan

KESIMPULAN DAN SARAN

Aplikasi analisis biomekanik ini dibuat untuk mengembangkan kemampuan berlari atlet. Untuk memenuhi kebutuhan tersebut, aplikasi ini dibuat menggunakan teknik pengukuran menggunakan vektor terhadap sebuah citra yang diambil dari hasil rekaman video performa atlet. Selain vektor, pengukuran juga dilakukan menggunakan perbandingan skala terhadap dua garis yang berbeda. Perbandingan skala ini berguna untuk melakukan pengukuran jarak pada citra yang sama. Jadi dapat disimpulkan bahwa pengukuran dengan vektor dan perbandingan skala sudah dapat digunakan untuk melakukan analisis biomekanik dalam proses pengembangan kemampuan berlari atlet.

Aplikasi ini memiliki keunggulan tersendiri yaitu berupa analisis tidak hanya dilakukan dengan pengukuran, akan tetapi juga dapat dengan melihat rekaman video performa atlet yang dimainkan secara *slow motion*. Selain itu, proses perekaman video performa atlet juga bisa dilakukan melalui aplikasi ini. Jadi sumber masukan data untuk aplikasi ini tidak hanya satu sumber saja.

Aplikasi yang dibuat sudah dapat memenuhi kebutuhan dan memecahkan masalah yang ada, akan tetapi aplikasi ini juga masih memiliki kekurangan. Kekurangan tersebut tidak bersifat fatal untuk saat ini, akan tetapi perkembangan juga dibutuhkan supaya aplikasi ini bisa terus digunakan.

Meninjau hasil penelitian ini, disarankan untuk melakukan pengembangan aplikasi ini sehingga dapat melakukan analisis dengan metode *image processing* seperti *movement analysis*. Jika hal ini dapat dilakukan, maka proses analisis akan menjadi optimal. Karena saat atlet dalam video bergerak maka analisis akan dilakukan secara otomatis.

DAFTAR PUSTAKA

- [1] Anonim. 2008. *Video*. <http://en.wikipedia.org/wiki/Video>
- [2] Ameerah, Leen. 2007. *Computer Graphics for Java Programmers Second Edition*. John Wiley & Sons.
- [3] Bartlett, Roger. 2002. *Introduction to Sports Biomechanics*. Taylor & Francis e-Library.
- [4] Bartlett, Roger. 2007. *Introduction to Sports Biomechanics Second Edition: Analysing Human Movement Pattern*. Taylor & Francis e-Library.
- [5] Fadlisyah, SSi. 2007. *Computer Vision dan Pengolahan Citra*. Yogyakarta: Penerbit ANDI.
- [6] Hill, F., S. 1990. *Computer Graphics Using Open GL Second Edition*. New Jersey: Prentice Hall.

- [7] Jaenudin. 2005. *Belajar Sendiri .Net dengan Visual C# 2005*. Yogyakarta: Penerbit ANDI.
- [8] Miano, John. 1999. *Compressed Image File Formats: JPEG, PNG, GIF, XBM, BMP*. Massachusetts: Addison Wesley Longman, Inc.
- [9] Munir, Rinaldi. 2004. *Pengolahan Citra Digital dengan Pendekatan Algoritmik*. Bandung: Penerbit INFORMATIKA.
- [10] Munawar. 2005. *Pemodelan Visual dengan UML*. Yogyakarta: Penerbit Graha Ilmu.
- [11] Rilman Adam Djamaris, Aurino. 2008. *Sistem Koordinat*. <http://mathematica.aurino.com/?tag=sistem-koordinat>
- [12] Serban, Iulian, dkk. 2006. *GDI+ Custom Controls with Visual C# 2005*. Birmingham: Packt.
- [13] Underdahl, Keith. 2003. *Digital Video For Dummies 3rd Edition*. New York: Wiley Publishing, Inc.
- [14] Wahyu Nurcahyo, Yohanes. 2007. *Sistem Koordinat*. <http://www.cekli.com/en/node/34>